CLAIMS

What is claimed is:

1.   A method for interprocess communication in a managed information architecture comprising:

5   receiving a registration from a service entity in the managed information architecture, the registration indicative of a significant occurrence in the managed information architecture and the service entity responsive to the significant occurrence;

establishing a persistent association of the service entity and the significant occurrence in response to the registration, the persistent association independent of the

10   enablement of the service entity, the persistent association providing a registered service entity;

receiving a notification indicative of the significant occurrence in the managed information architecture;

identifying, via the persistent association, the corresponding registered service

15   entity responsive to the significant occurrence;

enabling, if the identified registered service entity is disabled, a module including the service entity; and

invoking, via the persistent association, the service entity responsive to the significant occurrence.

20

2.   The method of claim 1 further comprising

detecting, via a class entity operable to execute instructions in the context of state information, the significant occurrence, and

transmitting an indication message indicative of the significant occurrence to a

25   module server operable to invoke the service entity.

3.   The method of claim 1 further comprising

disabling the module including the service entity; and

selectively enabling, in response to the significant occurrence, the module

30   including the service entity, wherein the persistent association is independent of enabling and disabling of the service entity.

4.   The method of claim 1 wherein, following selectively enabling:

enqueuing an indication of the significant occurrence a queue, the queue corresponding to the process including the module containing the service entity; and

5   assigning, to a thread corresponding to the queue, performance of the service entity corresponding to the significant occurrence.

5.   The method of claim 1 wherein the received registration employs a genericizing reference for identifying the service entity, the genericizing reference operable to avoid

10   extraneous references and further operable for registration of a plurality of service entities, each of the service entities independent of references of other of the plurality of service entities.

6.   The method of claim 1 wherein the invoking occurs in a different executable

15   entity than the significant occurrence and wherein the detecting further comprises transmitting the indication message from the process corresponding to the significant occurrence to the module including the service entity corresponding to the significant occurrence.

20   7.   The method of claim 1 wherein invoking further comprises:

identifying associated data indicative of the significant occurrence;

assembling an invocation call, the invocation call including a reference to the service entity and a reference to the identified associated data; and

executing the referenced service entity in the context of the referenced associated

25   data.

8.   The method of claim 7 wherein the executing further comprises a dispatch command, the dispatch command operative to enqueue multiple invocations to the same service entity, wherein the dispatch command references the associated data via a

30   genericizing reference, the genericizing reference operable to include multiple types of

associated data independently of the dispatched service entities employing the associated data.

9.   The method of claim 1 further comprising:

5   identifying, in a memory portion operable for dynamic allocation, an allocation adapted to store the notification indicative of the significant occurrence;

tracking, via an allocation manager operable to manage portions of dynamic memory, references to the allocation; and

deallocating, following execution of the service entity corresponding to the

10   significant occurrence, the allocation, the deallocation occurring in the same identified memory portion.

10.   The method of claim 1 wherein establishing the persistent association further comprises storing, in a global association table, an indication of the significant

15   occurrence and an indication of the module containing the service entity, the global association table persistently independent of enablement of the module including the service entity corresponding to the significant occurrence.

11.   The method of claim 1 wherein establishing the persistent association further

20   comprising storing, in a local association table, an indication of the significant occurrence and an indication of the service entity corresponding to the significant occurrence.

12.   The method of claim 1 wherein the service entities are handlers corresponding to executable methods and the indication messages are events propagated by an invocation

25   mechanism as a result of the significant occurrence service entity

13.   The method of claim 1 wherein associating an identity of the significant occurrence with a service entity occurs in a native language of the service entity and corresponding subscriber, and avoids a corresponding definition in an external interface

30   language, the external interface language for generating additional code, the additional code adapted for support and testing operations.

14.   The method of claim 13 wherein the external interface language is the Object Management Group Interface Definition Language (OMG/IDL).

5   15.   A method for invocation of subscribers comprising:

receiving a subscription associative of a service entity and a significant occurrence, the service entity having instructions operative for executing and completing a particular task upon an indication of the significant occurrence;

associating the significant occurrence with the service entity, the association

10   including a generic reference applicable to a plurality of service entities, the association further operable to selectively enable a module including the service entity upon the significant occurrence;

receiving the indication of the significant occurrence;

determining, via the association, the corresponding service entity and the module

15   including the service entity;

selectively enabling the module including the service entity; and

dispatching the service entity to execute and complete the time based task.

16.   The method of claim 15 wherein the associating is performed by an association

20   entry, the association entry further comprising a global entry and a local entry including an indication of the particular task.

17.   The method of claim 16 wherein the global entry is operable to trigger enablement of the module including the local entry if the module is not enabled upon the

25   notification of the significant occurrence.

18.   A method for interprocess communication in an information retrieval environment comprising:

defining an invocation message indicative of a significant occurrence in the

30   information retrieval environment, the invocation message corresponding to a common class associated with a plurality of invocation messages;

subscribing, from a local subscriber in a local process, to the invocation message for establishing reporting of the significant occurrence, subscribing further including specifying a service entity operable to process the invocation message;

registering an indication of the subscription in a local map operative for
5 associating significant occurrences and service entities in the local process for invoking the service entity responsively to an occurrence of the defined significant occurrence;

registering an indication of the subscription in a global map operative for associating invocation message with service entities, and further operable to invoke components including service entities in remote processes;

10 receiving a publication of the invocation message from a monitoring component;

propagating the publication and indexing the invocation message in the global map;

dispatching, based on the indexing, an indication of the publication to the local subscribers; and

15 invoking the component including the specified service entities in response to the dispatching and propagating the publication.

19.   The method of claim 18 wherein invoking further comprises selective activation of components including associated service entities, the service entities in inactive
20 components responding to the dispatch upon activation.

20.   The method of claim 19 wherein the components including the service entities need not be active during the publication, the inactive service entities being invoked accordingly in response to the dispatching, wherein unavailable service entities consume
25 fewer resources than available service entities.

21.   A services architecture for interprocess communication in a managed information system comprising:

a module server operable to receive a registration from a service entity in the
30 managed information system, the registration indicative of a significant occurrence in the

managed information system and the service entity responsive to the significant occurrence;

a service provider in the module server operable to establish a persistent association of the service entity and the significant occurrence in response to the registration, the persistent association independent of the enablement of the service entity, the persistent association providing a registered service entity for the service provided by the service entity;

an association mapping for storing the persistent association and adapted to map a received notification indicative of the significant occurrence in the managed information system, the service operable to identify, via the persistent association, the corresponding registered service entity responsive to the significant occurrence; and

an activation manager operable to enable, if the identified registered service entity is disabled, a module including the service entity, the module server operable to invoke, via the persistent association, the service entity responsive to the significant occurrence.

22.   The services architecture of claim 21 wherein the module server is in communication with a class entity adapted to detect the significant occurrence, and further operable to transmit an indication message indicative of the significant occurrence to the module server.

23.   The services architecture of claim 21 further comprising an activation manager, the activation manager operable to:

disable the module including the service entity; and

selectively enable, in response to the significant occurrence, the module including the service entity, wherein the persistent association is independent of enabling and disabling of the service entity.

24.   The services architecture of claim 21 wherein the module server is further operable to:

enqueue an indication of the significant occurrence a queue, the queue corresponding to the process including the module containing the service entity; and

assign, to a thread corresponding to the queue, performance of the service entity corresponding to the significant occurrence.

25.   The services architecture of claim 21 wherein the received registration includes a genericizing reference for identifying the service entity, the genericizing reference adapted to avoid extraneous references and further operable for registration of a plurality of service entities, each of the service entities independent of references of other of the plurality of service entities.

26.   The services architecture of claim 21 wherein the module server is further operable to invoke the service entity in a different executable entity than the significant occurrence, and further operable to transmit the indication message from the process corresponding to the significant occurrence to the module including the service entity corresponding to the significant occurrence.

27.   The services architecture of claim 21 wherein the module server is further operable to

identify associated data indicative of the significant occurrence;

assembling an invocation call, the invocation call including a reference to the service entity and a reference to the identified associated data; and

execute the referenced service entity in the context of the referenced associated data.

28.   The services architecture of claim 27 wherein module server is further operable to execute a dispatch command, the dispatch command operative to enqueue multiple invocations to the same service entity, wherein the dispatch command references the associated data via a genericizing reference, the genericizing reference operable to include multiple types of associated data independently of the dispatched service entities employing the associated data.

29.   The services architecture of claim 21 further comprising a memory/heap manager operable to identify, in a memory portion operable for dynamic allocation, an allocation adapted to store the notification indicative of the significant occurrence, the memory/heap manager further operable to tracking references to the allocation, and to deallocate, following execution of the service entity corresponding to the significant occurrence, the allocation, the deallocation occurring in the same identified memory portion.

30.   The services architecture of claim 21 further comprising a global association table operable to store the persistent association including an indication of the significant occurrence and an indication of the module containing the service entity, the global association table persistently independent of enablement of the module including the service entity corresponding to the significant occurrence.

31.   The services architecture of claim 21 further comprising storing a local association table adapted to store an indication of the significant occurrence and an indication of the service entity corresponding to the significant occurrence.

32.   The services architecture of claim 21 wherein the service entities are handlers corresponding to executable methods and the indication messages are events propagated by an invocation mechanism as a result of the significant occurrence Service entity

33.   A method for managing modules in a services infrastructure comprising:
        deploying a plurality of significant occurrences in the infrastructure environment;
        identifying service providers and user entities, the user entities operable for development and modification by a user, and the service providers unavailable for user modifications;
        identifying subscribers and publishers in the user entities, the subscribers having a service entity for handling a significant occurrence, and the publishers operable to detect the significant occurrences, generate clarifying data associated with the significant occurrence and publish a corresponding notification to the subscribers of the same significant occurrence with the corresponding associated data;

defining service entities operable to process each of the deployed significant occurrences referenced by the subscribers and publishers;

correlating each of the deployed significant occurrences with the corresponding subscribers and publishers; and

5   selectively invoking, upon publication of the significant occurrence by a publisher, each of the subscribers corresponding to the significant occurrence, the subscribers and publishers having knowledge only of the significant occurrence and associated clarifying data, the publication avoiding references by the user entities to the service providers processing the correlation of the significant occurrence with the
10   corresponding subscribers, such that the subscribers and included service entities are independently active from publishers of the corresponding significant occurrence.

34.   A computer program product having a computer readable medium operable to store computer program logic embodied in computer program code encoded thereon for
15   interprocess communication in a managed information architecture comprising:

computer program code for receiving a registration from a service entity in the managed information architecture, the registration indicative of a significant occurrence in the managed information architecture and the service entity responsive to the significant occurrence;

20   computer program code for establishing a persistent association of the service entity and the significant occurrence in response to the registration, the persistent association independent of the enablement of the service entity, the persistent association providing a registered service entity;

computer program code for receiving a notification indicative of the significant
25   occurrence in the managed information architecture;

computer program code for identifying, via the persistent association, the corresponding registered service entity responsive to the significant occurrence;

computer program code for enabling, if the identified registered service entity is disabled, a module including the service entity; and
30   computer program code for invoking, via the persistent association, the service entity responsive to the significant occurrence.

35.   A computer data signal having program code for thereon for interprocess communication in a managed information architecture comprising:

program code for receiving a registration from a service entity in the managed information architecture, the registration indicative of a significant occurrence in the managed information architecture and the service entity responsive to the significant occurrence;

program code for establishing a persistent association of the service entity and the significant occurrence in response to the registration, the persistent association independent of the enablement of the service entity, the persistent association providing a registered service entity;

program code for receiving a notification indicative of the significant occurrence in the managed information architecture;

program code for identifying, via the persistent association, the corresponding registered service entity responsive to the significant occurrence;

program code for enabling, if the identified registered service entity is disabled, a module including the service entity; and

program code for invoking, via the persistent association, the service entity responsive to the significant occurrence.

36.   A services architecture for interprocess communication in a managed information comprising:

means for receiving a registration from a service entity in the managed information architecture, the registration indicative of a significant occurrence in the managed information architecture and the service entity responsive to the significant occurrence;

means for establishing a persistent association of the service entity and the significant occurrence in response to the registration, the persistent association independent of the enablement of the service entity, the persistent association providing a registered service entity;

means for receiving a notification indicative of the significant occurrence in the managed information architecture;

means for identifying, via the persistent association, the corresponding registered service entity responsive to the significant occurrence;

5   means for enabling, if the identified registered service entity is disabled, a module including the service entity; and

means for invoking, via the persistent association, the service entity responsive to the significant occurrence.

10